OULUN SEUDUN
AMMATTIKORKEAKOULU

Arto Hurskainen

TIEDONSIIRTORAJAPINNAN TOTEUTUS MIFARE DESFIRE -TUNNISTEIDEN OHJELMOINTISOVELLUKSEEN

TIEDONSIIRTORAJAPINNAN TOTEUTUS MIFARE DESFIRE -TUNNISTEIDEN OHJELMOINTISOVELLUKSEEN

Arto Hurskainen
Opinnäytetyö
Kevät 2012
Tietotekniikan koulutusohjelma
Oulun seudun ammattikorkeakoulu

TIIVISTELMÄ

Oulun seudun ammattikorkeakoulu
Tietotekniikka, langaton tietoliikenne

Tekijä: Arto Hurskainen
Opinnäytetyön nimi: Tiedonsiirtorajapinnan toteutus Mifare DESFire -tunnisteiden ohjelmointisovellukseen
Työn ohjaaja: Kari Jyrkkä
Työn valmistumislukukausi ja -vuosi: Kevät 2012 Sivumäärä: 42 + 0

Opinnäytetyön lähtökohtana toimi Idesco Oy:n tuotannossa oleva RFID-tunnisteita valmistava automaatiorobotti ja siihen lisättävä tuki Mifare DESFire -etätunnistustekniikalle. Työn tehtävänä oli toteuttaa tiedonsiirtorajapinta robotin ja robottia ohjaavan käyttöliittymän välille. Työn lopputuloksena oli tarkoitus olla valmis kokonaisuus, jota voitiin käyttää Mifare DESFire -tunnisteiden valmistamisessa.

Työ aloitettiin listaamalla tiedonsiirtorajapinnan vaatimukset, jonka jälkeen ryhdyttiin tutustumaan Idescon automaatiorobotin toimintalogiikkaan, Mifare DESFire -etätunnistustekniikkaan, Linx-leimasimeen sekä jo olemassa oleviin robotin käyttöliittymiin. Tämän jälkeen rajapinta ohjelmoitiin C++-ohjelmointikielellä. Lopuksi ohjelmoitu rajapinta liitettiin osaksi käyttöliittymäsovellusta ja käyttöliittymäsovelluksen toiminta testattiin.

Työn tuloksena saatiin toimiva kokonaisuus, jonka avulla Mifare DESFire -tunnisteita pystytään valmistamaan tehokkaammin. Tuotetta käytetään jatkossa Idescon tuotannossa.

Asiasanat: RFID, Mifare, etätunnistus, ohjelmointi, robotit

ABSTRACT

Oulu University of Applied Sciences
Information Technology, Wireless Telecommunication

Author: Arto Hurskainen
Title of thesis: Implementation of the communication interface to programming application of Mifare DESFire identifiers
Supervisor: Kari Jyrkkä
Term and year when the thesis was submitted: Spring 2012 Pages: 42 + 0

The purpose of this thesis was to add a Mifare DESFire support to the Idesco's manufacturing robot which makes RFID transponders. My job was to create communication interface between a robot and the user interface which controls the robot.

The job was begun by listing the requirements of the communication interface. After that I became acquainted with the operation logic of the robot, Mifare DESFire technology, Linx printer and the existing user interfaces for the robot. Then the programming of the interface was begun with a C++ programming language. Finally the programmed interface was connected as part of the user interface and was tested with the help of the robot.

As a result of this project we got a functional system which can be used to manufacture Mifare DESFire transponders more efficiently. The system will be used in Idesco's production.

Keywords: RFID, Mifare, remote sensing, programming, robots

ALKULAUSE

Tämä opinnäytetyö on tehty syksyn 2011 ja alkuvuoden 2012 välisenä aikana Idesco Oy:lle. Tehtävänä oli toteuttaa tiedonsiirtorajapinta Mifare DESFire -tunnisteiden ohjelmointisovellukseen.

Haluan kiittää Idesco Oy:n tuotekehitysjohtajaa Anu-Leena Arolaa, jolta sain opinnäytetyöni aiheen. Kiitokset insinööriopiskelija Ville Koivumäelle hyvin sujuneesta yhteistyöstä projektin aikana. Työn valvojalle lehtori Kari Jyrkälle haluan esittää kiitokset opinnäytetyöni ohjauksesta. Lopuksi haluan vielä kiittää muita Idescon henkilökuntaan kuuluvia, jotka auttoivat tämän opinnäytetyön onnistumisessa.

Oulussa 16.2.2012

Arto Hurskainen

SISÄLLYS

SANASTO

3DES	Triple DES, DES-salauksen kehittyneempi versio
AES	Advanced Encryption Standard, lohkosalausmenetelmä
AID	Application Identifier, Mifare DESFire -tunnisteen sovellustunnus
CRC	Cyclic Redundance Check, tiivistealgoritmi
DES	Data Encryption Standard, lohkosalausmenetelmä
DLL	Dynamic Link Library, jaettu ohjelmakirjasto
DSR	Data Set Ready, RS-232-väylän signaali
DTR	Data Terminal Ready, RS-232-väylän signaali
GUI	Graphical User Interface, graafinen käyttöliittymä
HF	High Frequency, taajuusalue välillä 3–30 MHz
LF	Low Frequency, taajuusalue välillä 30–300 kHz
MAC	Message Authentication Code, autentikointimenetelmä
PICC	Proximity Integrated Circuit Card, RFID-tunniste
PLC	Programmable Logic Controller, ohjelmoitava logiikka
RFID	Radio Frequency Identification, radiotaajuinen etätunnistus
RS232	Recommended Standard 232, sarjamuotoinen tiedonsiirtostandardi
UHF	Ultra High Frequency, taajuusalue välillä 0,3–3 GHz
UID	Unique Identification, RFID-tunnisteen yksilöllinen sarjanumero

1 JOHDANTO

Mifare DESFire on 13,56 MHz:n taajuudella toimiva etätunnistetekniikka. Tekniikka käyttää joustavaa muistirakennetta, jonka avulla samaa tunnistetta voidaan käyttää useaan eri sovellukseen. Mifare DESFiren tietoturvaominaisuuksien ansiosta etätunnistus pystytään suorittamaan turvallisesti.

Idesco Oy on vuonna 1989 perustettu oululainen RFID-laitteita sekä -tunnisteita valmistava yritys. Idescon monipuolisiiin teknologioihin perustuvia tuotteita käytetään esimerkiksi henkilön- ja ajoneuvontunnistuksessa sekä kulunvalvonnassa.

Mifare DESFire -tunnisteiden kovan kysynnän vuoksi Idesco Oy tarjosi opinnäytetyön aihetta, jossa Idescon tuotannossa olevaan automaatiorobottiin ryhdyttiin toteuttamaan käyttöliittymää, jolla voitiin automatisoidusti ohjelmoida Mifare DESFire -tunnisteita asiakkaiden tarpeisiin.

Käyttöliittymän toteuttaminen jakautui kahteen osaan: tiedonsiirtorajapinnan toteutus sekä käyttöliittymän käyttäjälle näkyvän osion toteutus. Tehtäväni oli toteuttaa tiedonsiirtorajapinta ja Ville Koivumäki toteutti omana opinnäytetyönään toisen osan.

2 RADIOTAAJUINEN ETÄTUNNISTUS

Radiotaajuinen etätunnistus eli RFID (Radio Frequency Identification) tarkoittaa radiotaajuuksilla toimivia tekniikoita, joita käytetään muun muassa tuotteiden ja asioiden havainnointiin, yksilöintiin ja tunnistamiseen. Teknologian toiminta perustuu tiedon tallentamiseen RFID-tunnisteeseen ja sen sisällön lukemiseen RFID-lukijalla radioaaltojen avulla. (1.)

RFID-tekniikkaa voidaan pitää perinteisen viivakoodin korvaajana. Erona viivakoodiin on se, että RFID-tunniste voidaan lukea ilman suoraa katsekontaktia. Lisäksi RFID-tunnisteiden sisältöä voidaan muuttaa, kun taas viivakoodi pysyy muuttumattomana. RFID-tunnisteen toiminta ei keskeydy vaikeissakaan olosuhteissa, kun taas viivakoodi on hyvin altis esimerkiksi likaantumiselle. (1.)

2.1 Tekniikan perusteet

RFID-tekniikka sisältää kolme eri tunnistetyyppiä: passiiviset RFID-tunnisteet, puolipassiiviset RFID-tunnisteet ja aktiiviset RFID-tunnisteet (kuva 1).

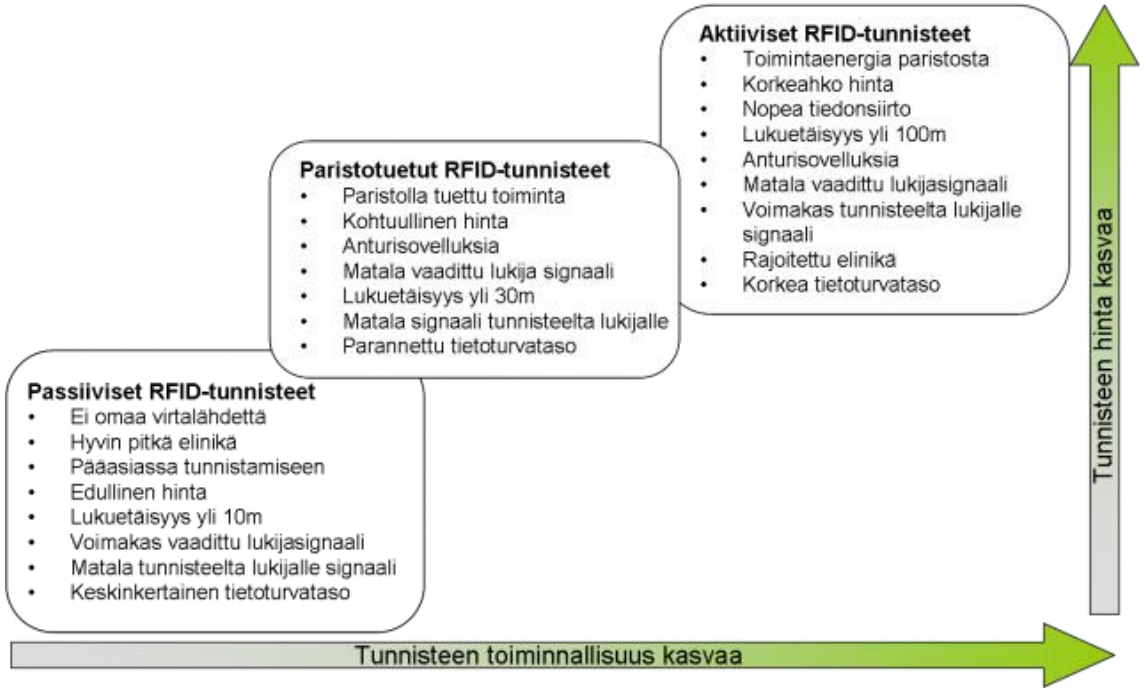
Yksinkertaiset, passiiviset tunnisteet eivät sisällä omaa virtalähdettä, vaan ne saavat toimintaansa tarvittavan tehon lukijalaitteelta. Tämä tarkoittaa sitä, että esimerkiksi tietoa ei voida kirjoittaa tunnisteeseen, ellei tunniste pysy koko toiminteen ajan lukijan magneettikentässä. (2, s. 10.)

Puolipassiiviset tunnisteet sisältävät oman virtalähteen, mutta sitä käytetään vain tietojen lähettämiseen lukijalle, kun ensin on vastaanotettu lukijan lähettämä signaali. Tällä saavutetaan passiivisiin tunnisteisiin verrattuna pitempi lukuetäisyys. Muuten puolipassiivisen tunnisteen toiminnallisuus on passiivisen tunnisteen kaltainen. (2, s. 10–11.)

Aktiiviset tunnisteet sisältävät oman virtalähteen ja niissä sitä voidaan käyttää myös tunnisteen laskennan virtalähteenä. Näin ollen tunnisteeseen on mahdollista kirjoittaa tietoa myös silloin, kun se ei ole lukijan lukuetäisyydellä. Aktiiviseen tunnisteeseen voidaan liittää esimerkiksi lämpötila-anturi, jolloin

tunniste voi tietyin väliajoin käydä lukemassa lämpötilatietoa anturilta ja lähettää lukemansa tiedon eteenpäin saavuttuaan lukijan lukuetäisyydelle. (2, s. 11.)

Passiivisten tunnisteiden hinnat liikkuvat kymmenissä senteissä, kun taas aktiiviset tunnisteet voivat maksaa jopa euron (2, s. 10).



KUVA 1. Passiivisten, paristotuettujen (puolipassiivisten) ja aktiivisten RFID-tunnisteiden yleispiirteet (3)

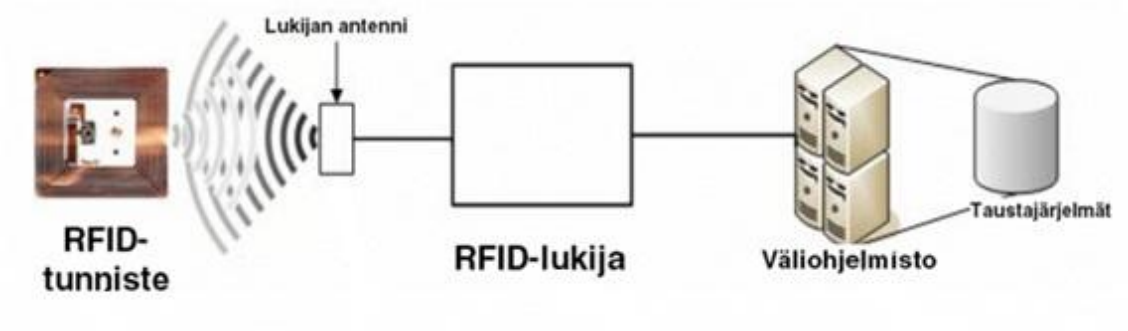
RFID-laitteilla on kaksi eri toimintamenetelmää käytettäessä passiivisia tunnisteita (3).

LF (Low Frequency)- ja HF (High Frequency) -taajuusalueilla RFID-lukija ja -tunniste muodostavat keskenään induktiivisen kytkennän. Lukija luo oskilloivan magneettikentän johtamalla vaihtovirtaa antenniinsa. Magneettikenttä indusoi vastaavan vaihtovirran tunnisteen antenniin, mikäli se sijaitsee tarpeeksi lähellä lukijaa. RFID-tunnisteen siru saa toimintakykynsä indusoituneesta vaihtovirrasta. Tunnisteen saatua toimintakykynsä sirun EEPROM-muistissa olevan datan avulla moduloidaan tunnisteen antennissa olevaa virtaa, joka näkyy magneettikentän yli lukijan antennin jännitteessä. (3.)

UHF (Ultra High Frequency) -taajuusalueella tunniste ja lukija keskustelevat keskenään radioaaltoja välittämällä, samaan tapaan kuin esimerkiksi matkapuhelimet ja radiot. RFID-lukija lähettää antennillaan radioaaltoja, tunniste vastaanottaa radioaallot ja heijastaa ne takaisin sirun muistissa olevan tiedon kera. (3.)

2.2 RFID-järjestelmän rakenne

RFID-järjestelmään kuuluu RFID-tunniste ja -lukija sekä jonkinlainen taustajärjestelmä tai -sovellus (kuva 2) (3).



KUVA 2. RFID-järjestelmän rakenne (3)

RFID-tunniste koostuu antennista ja mikrosirusta, ja tunnisteet ovat yleensä kortin, napin, tarran tai implantin muodossa. Sirun muistissa on kiinteä yksilöllinen sarjanumero ja lisäksi standardin mukaisesti vapaata kirjoitustilaa. RFID-lukija lukee tietoa tunnisteesta, ja osassa laitteita se pystyy myös kirjoittamaan tietoa tunnisteeseen. Käytettäessä passiivisia tunnisteita lukija antaa tunnisteelle tiedonsiirtoon tarvittavan energian. Taustajärjestelmä vastaanottaa ja käsittelee lukijoilta saadun tiedon, huolehtii tiedonjaosta ja voi hallinnoida lukijoita. (3.)

2.3 Taajuusalueet ja käyttökohteet

RFID-teknologia sisältää useita eri taajuusalueita ja niitä käytetään erilaisiin sovelluksiin. Kullakin taajuusalueella on omat erityispiirteensä, jotka vaikuttavat muun muassa lukuetäisyyteen ja läpäisykykyyn. Lukijassa ja tunnisteessa käytettävän antennin koko riippuu käytetystä taajuudesta, ja näin ollen taajuus

rajoittaa mahdollisuuksia pienentää sekä lukijan että tunnisteen fyysistä kokoa. (2, s. 8.)

LF-taajuusalueella RFID-järjestelmien yleisin taajuus on 125 kHz. LF-järjestelmien käyttö on vähäistä uusissa sovelluskohteissa, ja niitä käytetään lähinnä kulunvalvonnassa ja eläintunnistuksen sovelluksissa. (4.) Taajuusalueen heikkoutena voidaan pitää lyhyttä lukuetäisyyttä. Hyvänä puolena on se, että LF-taajuusalueen RFID-tunnisteet toimivat hyvin metallien läheisyydessä. (2, s. 9.)

HF-taajuuksilla käytössä on 13,56 MHz:n taajuus. HF-järjestelmiä käytetään paljon kulunvalvonnassa. HF-alueen tunnisteiden heikkoutena on lyhyt lukuetäisyys sekä niiden huono toimivuus metallien läheisyydessä. Tekniikan mahdollisuuksia ovat sen laaja käytettävyys sekä laaja valikoima erilaisia tunnisteita. (2, s. 9.)

UHF-taajuusalueen järjestelmissä taajuudet vaihtelevat hieman ympäri maailmaa. Euroopassa käytetään 869 MHz:n taajuutta. UHF-järjestelmiä käytetään paljon erilaisissa logistiikan sovelluksissa. (4.) UHF-taajuusalueen RFID-tekniikalla on mahdollista saavuttaa jopa 1–5 metrin lukuetäisyys. Taajuusalueen heikkoutena on sen huono toiminta vedessä tai vettä sisältävässä aineessa. Tekniikan käyttöä rajoittaa myös se, että eri puolilla maailmaa käytetään eri taajuutta. (2, s. 9.)

Mikroaaltoalueen RFID-järjestelmissä käytetään 2,4 GHz:n taajuutta. Tunnetuimpia sovelluskohteita on esimerkiksi tietullin automaattinen tunnistus. Mikroaaltojärjestelmissä käytetään yleensä aktiivisia RFID-tunnisteita. (4.) Mikroaaltoalueen hyvinä puolina voidaan pitää sitä, että tunniste on mahdollista tehdä hyvin pienikokoiseksi. Heikkoutena myös tällä taajuusalueella on se, että se ei toimi kunnolla veden läheisyydessä. (2, s. 9.)

3 MIFARE-TUOTTEET

Mifare on NXP Semiconductorsin omistuksessa oleva tuotemerkki, joka käsittää kontaktittomissa etätunnisteissa sekä etätunnistimien lukijalaitteissa käytettäviä mikropiirejä. Mifare-tuoteperheeseen kuuluu kuusi erilaista etätunnistetekniikkaa, jotka perustuvat 13,56 MHz:n ISO/IEC 14443 type A -standardin eri osiin (taulukko 1). Jokainen tekniikka pitää sisällään sekä tunnisteen että tunnistetta lukevan lukijan piirit ja niiden määritykset. (5.) Tässä työssä ei käsitellä kaikkia eri Mifare-tekniikoita, vaan paneudutaan ainoastaan Mifare DESfire -tekniikkaan.

TAULUKKO 1. Mifare-etätunnistetekniikoiden ominaisuudet (6, s. 5)

	MIFARE Ultralight	**MIFARE Ultralight C**	**MIFARE Classic**	**MIFARE Plus**	**MIFARE DESFire**	**DIF (like SmartMX)**
HW Crypto	-	3DES	Crypto1	Crypto1, AES	3DES, AES	3DES, AES, PKE
EEPROM	512 bit	1536 bit	320 Bytes, 1k Bytes, 4k Bytes	2k Bytes, 4k Bytes	2k Bytes, 4k Bytes, 8k Bytes	4k Bytes – 144k Bytes
Special Features	-	-	-	MIFARE Classic compatible	-	MIFARE Classic compatible
Certification	-	-	-	CC EAL 4+	CC EAL 4+	CC EAL 5+
Contactless interface	ISO/IEC 14443A	ISO/IEC 14443A	ISO/IEC 14443A	ISO/IEC 14443A	ISO/IEC 14443A	ISO/IEC 14443A

3.1 ISO/IEC 14443 -standardi

ISO/IEC 14443 -standardi kuvaa kontaktittoman etätunnisteen (PICC) toimintatavan ja toimintaan liittyvät parametrit. Standardi ei käsitä kaikkia etätunnisteita, vaan ainoastaan noin 7–15 cm:n etäisyydellä RFID-lukijasta toimivat tunnisteet. Standardi koostuu neljästä osasta: osa 1 fyysiset ominaisuudet, osa 2 RF-teho ja signaalirajapinta, osa 3 alustus ja törmäyksien välttäminen ja osa 4 siirtoprotokolla. (7, s. 240.)

Osassa 1 kuvataan etätunnisteen fyysiset ja mekaaniset ominaisuudet, esimerkiksi etätunnisteen mitat. Tämän lisäksi tässä osassa kerrotaan, kuinka

paljon etätunnistetta kärsii taivuttaa ja vääntää sekä miten tunniste käyttäytyy ultravioletti-, röntgen- ja sähkömagneettisen säteilyn alaisuudessa. (7, s. 240.)

Standardin osassa 2 kerrotaan, että etätunniste saa toimintaenergiansa lukijalaitteen luomasta magneettikentästä, jonka taajuutena on 13,56 MHz. Osassa 2 on kuvattu myös se, että lukijan tuottaman magneettikentän pitää pysyä tarkoin määritettyjen rajojen sisällä. Lukijan ja etätunnisteen välinen datan siirto, modulaatio ja käytetyt koodaustavat kuvataan myös tässä osiossa. ISO/IEC 14443 -standardissa on kaksi eri kommunikointitapaa: tyyppi A ja tyyppi B. Kommunikointitavat eroavat toisistaan standardin tässä osiossa lähinnä modulaation ja koodausmenetelmän mukaan. (7, s. 240–243.)

Alustus ja törmäyksien välttäminen -osiossa kuvataan, mitä tapahtuu, kun etätunniste viedään lukijan magneettikenttään, ja mitä tapahtuu, jos lukijan magneettikentässä on useampi tunniste. Standardissa kerrotaan tarkkaan, miten tunnisteen ja lukijan välinen yhteyden alustus etenee ja millaisia parametreja niiden välillä liikkuu sekä miten toimitaan, jos useamman tunnisteen lähettäessä syntyy törmäys. Tämä osio käsittelee myös kaksi eri menettelytapaa, tyypin A ja tyypin B. (7, s. 245–248.)

Kun lukijan ja etätunnisteen välille on saatu muodostettua yhteys, voidaan aloittaa datan lukeminen tunnisteelta ja datan kirjoittaminen tunnisteelle. ISO/IEC 14443 -standardin osassa 4 kerrotaan datansiirtoprotokollan rakenne, välitettävät parametrit ja siirron aikana syntyvien virheiden käsittely. (7, s. 251–255.)

3.2 Mifare DESFire

Mifare DESFire on etätunnistetekniikka, joka tarjoaa hyvät tietoturvaominaisuudet. Tekniikasta on olemassa kahta erilaista variaatiota, perus-DESFire sekä uudempi versio, DESFire EV1. (5.) Tässä luvussa käsitellään pelkästään uudempaa DESFire EV1 -tekniikkaa.

Mifare DESFire EV1:n avulla voidaan yhdistää useita eri sovelluksia yhdelle tunnisteelle ja se mahdollistaa luotettavan datansiirron tunnisteen ja lukijan

välillä. Mifare DESFire EV1 tukee kaikkia neljää ISO/IEC 14443A -standardin osaa. (8.) Taulukossa 2 on kuvattu DESFire EV1:n tärkeimpiä ominaisuuksia.

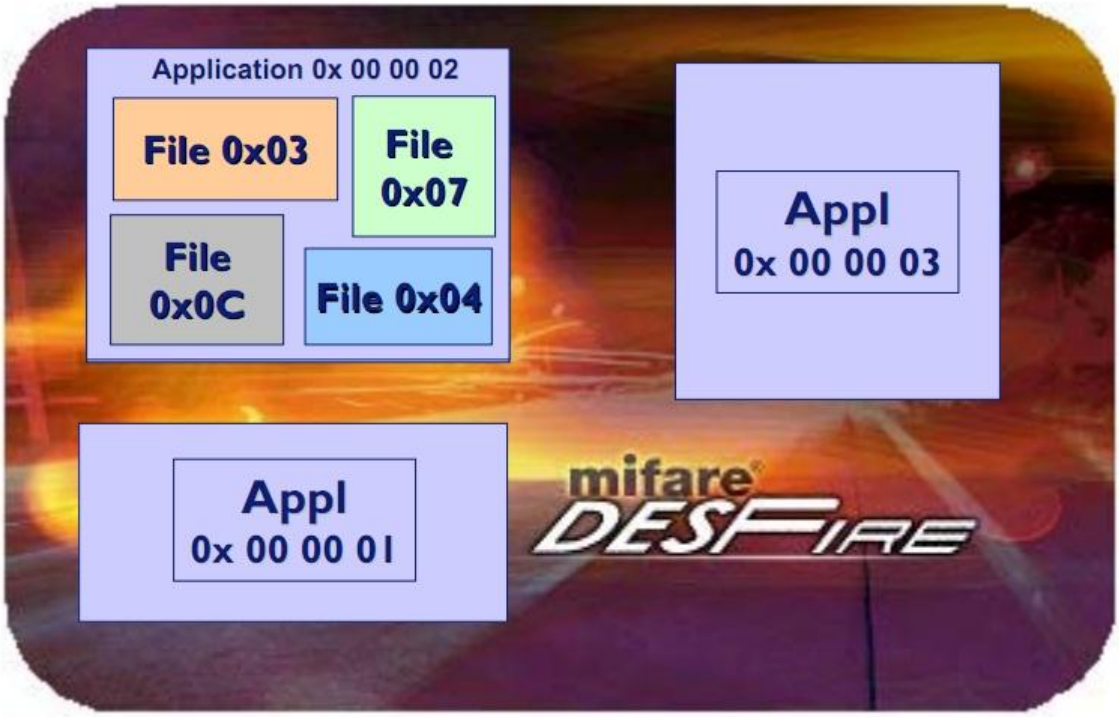
TAULUKKO 2. Mifare DESFire EV1:n ominaisuudet (8)

Product features	**Mifare DESFire EV1 2k**	**Mifare DESFire EV1 4k**	**Mifare DESFire EV1 8k**
Memory			
Eeprom Size [byte]	2048	4096	8192
Write endurance	500 000	500 000	500 000
Data redention	10	10	10
Organization	flexible file system	flexible file system	flexible file system
RF interface			
Acc. to ISO 14443 A	Yes - up to layer 4	Yes - up to layer 4	Yes - up to layer 4
Frequency [MHz]	13.56	13.56	13.56
Baudrate [kbit/s]	106 ... 848	106 ... 848	106 ... 848
Anticollision	bit-wise	bit-wise	bit-wise
Operating distance [mm]	up to 100	up to 100	up to 100
Security			
Unique serial number [byte]	7, cascaded	7, cascaded	7, cascaded
Random number generator	yes	yes	yes
Access keys	14 keys per application	14 keys per application	14 keys per application
Access conditions	per file	per file	per file
DES & 3DES security	MACing/encipherment	MACing/encipherment	MACing/encipherment
AES security	MACing/encipherment	MACing/encipherment	MACing/encipherment
Anti-tear supported by chip	yes	yes	yes
Special features			
Multi-application	28 application, MAD3	28 application, MAD3	28 application, MAD3
Purse functionality	value file	value file	value file
Transaction logging capability	record file	record file	record file
Secure transport transaction example	512 byte read	512 byte read	512 byte read
	128 byte write	128 byte write	128 byte write
Related transaction time [ms]	89	89	89

Tekniikkaa on saatavilla kolmella eri muistin koolla: 2 kB, 4 kB tai 8 kB. Mifare DESFiressä voidaan käyttää tiedon salaamiseen DES-, 3DES- tai AES-salausta ja jokainen Mifare DESFire-tunniste sisältää 7 tavua pitkän yksilöllisen sarjanumeron (UID). (8.)

3.3 Mifare DESFire -tunnisteen muistirakenne

Mifare DESFire -tunnisteen muisti koostuu sovelluksista ja tiedostoista. Yhteen tunnisteeseen voidaan luoda 28 sovellusta ja jokainen sovellus voi sisältää 32 tiedostoa (kuva 3). Jokainen sovellus tunnistetaan 3 tavua pitkällä AID:lla (Application IDentifier). (9, s. 7.)



KUVA 3. DESFire-tunniste koostuu sovelluksista ja tiedostoista (10, s.18)

Tiedostojen nimeämiseen käytetään tavun mittaista tunnistetta ja se määritetään tiedoston luonnin yhteydessä. Tiedoston luonnissa määritetään myös tiedoston koko ja tiedoston tyyppi. DESFire-tunniste voi sisältää viisi erilaista tiedostotyyppiä: standard data-, back-up data-, value-, linear record- ja cyclic record -tiedosto. Standard data- ja back-up data -tiedostoja voidaan käyttää esimerkiksi kortinhaltijan tietojen ja datan säilyttämiseen. Valuetiedostoon voidaan asettaa jokin arvo, jota voidaan joko kasvattaa (credit) tai pienentää (debit) asetettuihin raja-arvoihin. Linear record- ja cyclic record -tiedostoja voidaan käyttää samankaltaisen datan varastointiin. (9.) Jokaiselle tiedostolle voidaan määrittää neljä eri pääsyoikeutta: luku-, kirjoitus-, luku-kirjoitus- sekä muutosoikeus (kuva 4).

	Value File				all other files
R	Get Value	Debit			Read
W	Get Value	Debit	Limited Credit		Write
R&W	Get Value	Debit	Limited Credit	Credit	Read&Write
C	Change Config				

KUVA 4. Tiedoston pääsyoikeudet (10, s.16)

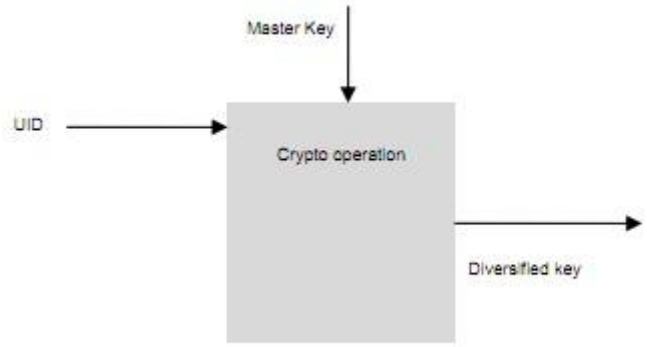
3.4 Mifare DESFiren turvallisuus

Kun tunniste saapuu lukijan magneettikenttään, ei voida olettaa, että se välttämättä kuuluu tai että sillä on oikeutta toimia käytetyssä sovelluksessa. Tämä voidaan selvittää käyttämällä jotakin autentikointitapaa.

Mifare DESFiressä voidaan tehdä kahdenkeskinen kolmivaiheinen autentikointi (9, s. 8). Kahdenkeskisessä kolmivaiheisessa autentikoinnissa sekä lukijalla että tunnisteella on tiedossa sama avain. Aluksi lukija lähettää tunnisteelle GET_CHALLENGE-komennon (kuva 5). Tämän jälkeen tunniste generoi satunnaisluvun R_a ja lähettää sen lukijalle. Lukija generoi satunnaisluvun R_b. Seuraavaksi lukija laskee käytetyn avaimen ja sovitun algoritmin avulla salatun datapaketin, joka sisältää satunnaisluvut R_a ja R_b. Datapaketti lähetetään tunnisteelle, tunniste purkaa paketin avaimellaan ja vertaa paketissa ollutta R_a-satunnaislukua aikaisemmin luomaansa satunnaislukuun. Jos nämä luvut täsmäävät, tunniste tietää, että molemmilla osapuolilla on samat avaimet. Tässä vaiheessa lukija ei vielä ole varma, että molemmilla on samat avaimet. Tunniste generoi toisen satunnaisluvun R_{a2}, laskee datapaketin, jossa on satunnaisluvut R_{a2} ja R_b, ja lähettää sen lukijalle. Lukija purkaa paketin ja vertaa paketissa ollutta satunnaislukua R_b aikaisemmin luomaansa satunnaislukuun. Jos luvut ovat samat, myös lukija varmistuu, että molemmilla on samat avaimet. (7, s. 222.)

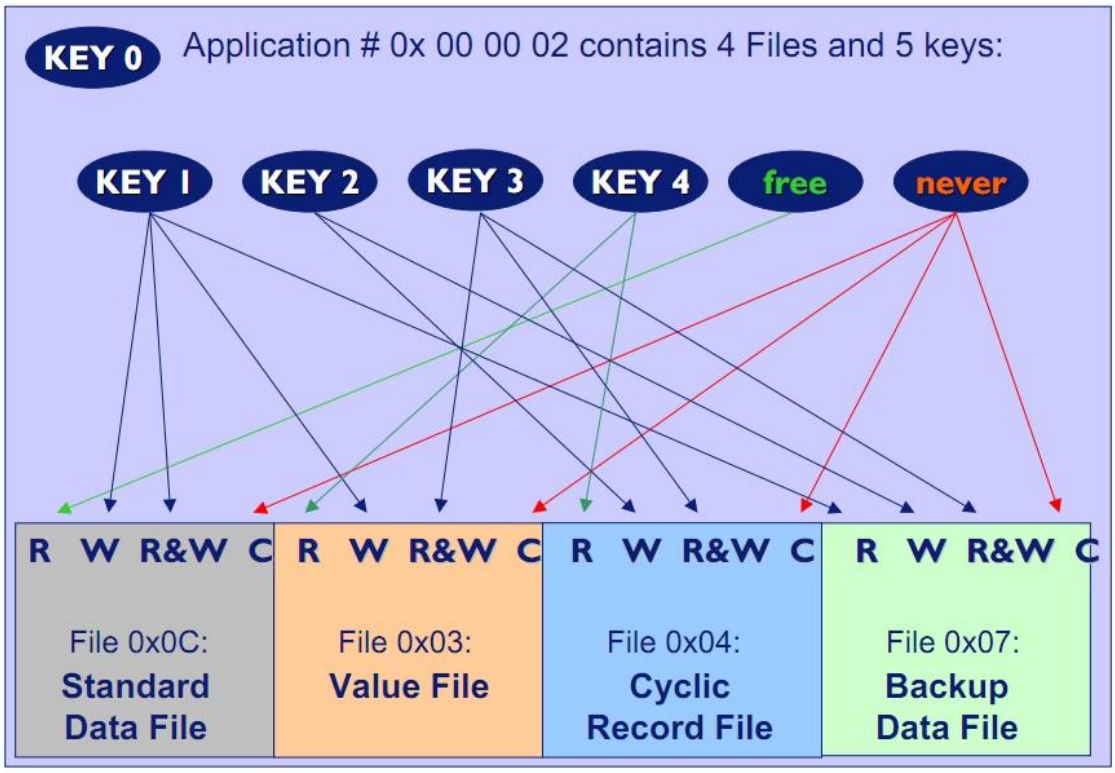
KUVA 5. Kahdenkeskinen kolmivaiheinen autentikointi (7, s. 222)

Mifare DESFire -tekniikassa turvallisuutta voidaan parantaa käyttämällä monipuolisia avaimia. Avaimen monipuolistamisen periaate on se, että yhdelläkään tunnisteella ei ole samaa avainta. Jokaisella tunnisteella on oma yksilöllinen UID, ja kun UID:lle ja tunnisteen alkuperäiselle avaimelle suoritetaan krypto-operaatio, tuloksena on monipuolistettu avain (kuva 6). (11, s. 8.)



KUVA 6. Monipuolisen avaimen luominen (11)

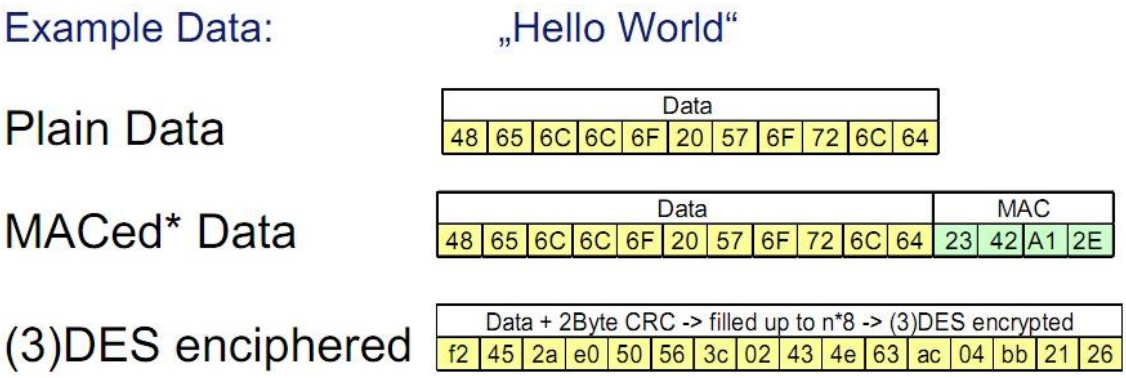
DESFire-tunnisteessa voidaan jokaiselle sovellukselle antaa käyttöön 14 eri avainta ja jokaisella tiedoston pääsyoikeudella voi olla käytössään eri avain (kuva 7). Tämän lisäksi tiedoston pääsyoikeudet voidaan asettaa niin, että niitä päästään käyttämään ilman autentikointia tai että niitä ei voida käyttää ollenkaan. Avain numero 0:aa käytetään tunniste- ja sovellustason hallinnoimiseen. (10.)



KUVA 7. Eri avainten käyttö eri pääsyoikeuksilla (10, s. 20)

Kun lukija ja tunniste ovat varmoja toistensa aitoudesta, voidaan aloittaa datansiirto. Lukijan ja tunnisteen välinen datansiirto voidaan tehdä kolmella eri tavalla:

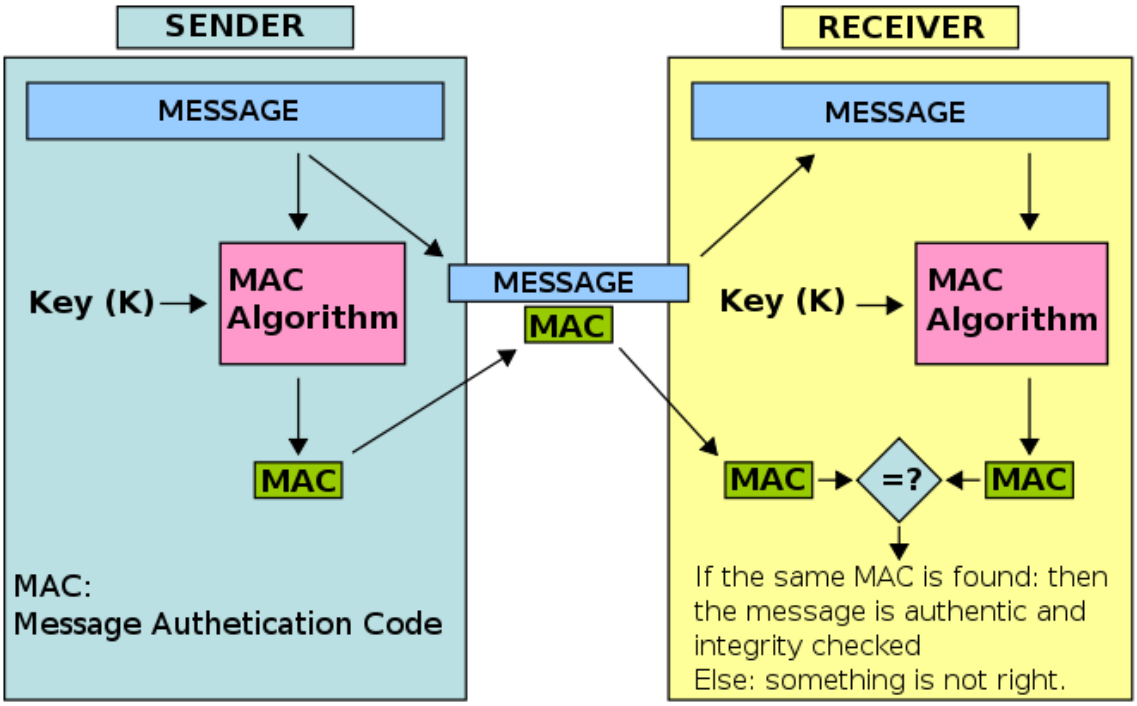
- datansiirto ilman salausta
- salaamaton data + datan autentikointikoodi (MAC)
- täysin salattu datansiirto (DES-, 3DES- tai AES-salauksella) + CRC-tarkiste (kuva 8). (9, s. 8.)



KUVA 8. DESFire-tekniikan datansiirtomenetelmät (10, s. 14)

MAC-autentikointi

Message Authentication Code eli MAC on tiivistefunktio, jota käytetään datan autentikoimiseen. Viestin lähettäjä syöttää MAC-algoritmiin salaisen avaimen sekä mielivaltaisen mittaisen viestin ja tuloksena saadaan MAC (kuva 9). Vastaanottajalla täytyy tietää salainen avain, jotta se voi laskea MAC-arvon. Jos vastaanotetun viestin MAC-arvo sekä vastaanottajan laskema MAC-arvo ovat samat, viesti on aito. (12.)



KUVA 9. MAC-algoritmin toiminta (12)

DES- ja 3DES-salaus

Data Encryption Standard (DES) on lohkosalausmenetelmä, joka vastaanottaa selväkielisen datan ja luo saman mittaisen salatun datan. DES-algoritmissa käytetään symmetristä avainta, mikä tarkoittaa sitä, että sekä salaaja että purkaja käyttävät samaa avainta. Salausavaimen pituus on 56 bittiä, jota voidaan pitää melko heikkona. (13, s. 2.)

3DES eli Triple DES on DES-salauksen kehittyneempi versio. 3DES suorittaa DES-algoritmin kolme kertaa peräkkäin ja näin avaimen pituudeksi saadaan 168 bittiä. (13, s. 5.)

AES-salaus

AES (Advanced Encryption Standard) on vuonna 2000 standardoitu salausmenetelmä. Se on toiminnaltaan erittäin nopea ja yksinkertainen, symmetrisellä avaimella toimiva lohkosalausmenetelmä. Salausmenetelmä tukee 128-, 192- ja 256-bittisiä avaimia. AES:n uskotaan pystyvän vastustamaan kaikkia tunnettuja lohkosalaimia vastaan tehtyjä hyökkäyksiä. Tämän lisäksi turvallisuutta tuo se, että AES:ssa ei esiinny ollenkaan heikkoja salausavaimia, mikä on ongelmana DES-salauksessa. (14.)

3.5 Mifare DESFiren käyttökohteita

Mifare DESFire -tunnisteen joustavan muistirakenteen ansiosta samalle tunnisteelle on mahdollista asettaa useita eri sovelluksia ja, mikä parasta, se on vieläpä turvallista.

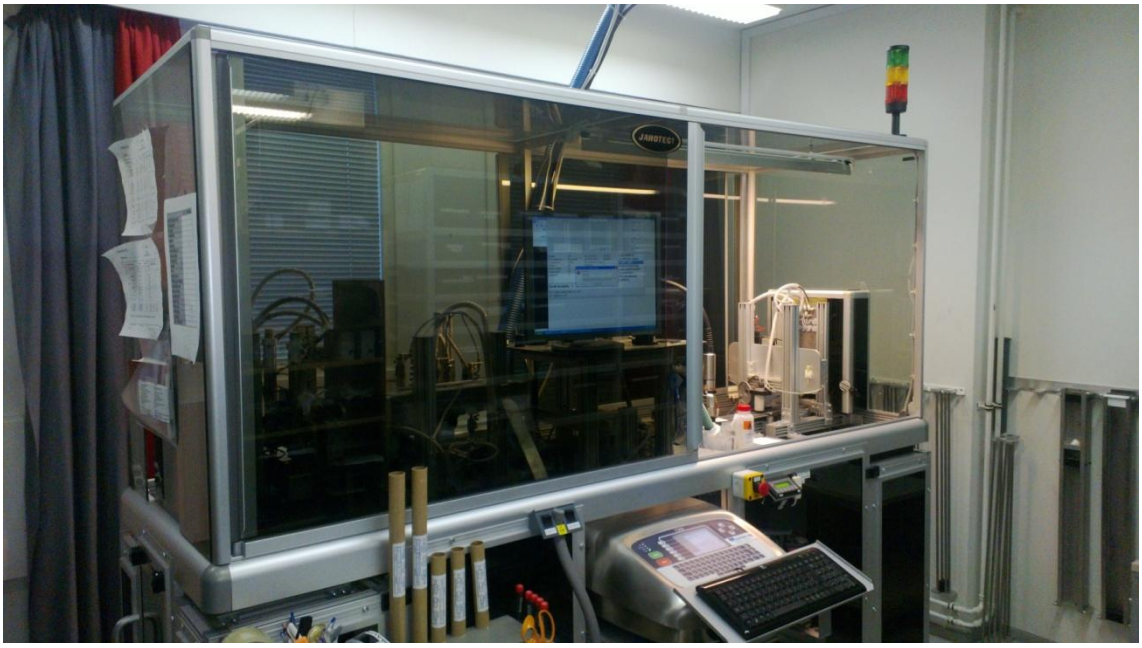
Elektronista lippujärjestelmää voidaan tänä päivänä pitää yhtenä suurimmista sovelluskohteista. Lataamalla lippuja tunnisteelle voidaan sitä käyttää maksuvälineenä esimerkiksi julkisen liikenteen palveluissa. Käyttäjän tarvitsee ainoastaan vilauttaa tunnistetta lukijaan maksaessaan matkaa. Tällä tavoin säästetään huomattavasti aikaa ja käyttäjällä ei tarvitse välttämättä olla edes rahaa mukana, ja hän voi silti matkustaa julkisilla kulkuneuvoilla. Tunniste voidaan ohjelmoida toimimaan sovelluksessa jonkin tietyn ajan verran tai johonkin tiettyyn päivämäärään asti tai siihen voidaan asettaa jokin tietty lukumäärä matkoja. (15.) Samaa periaatetta voidaan käyttää muun muassa tietullin maksamisessa sekä kuntosali- ja uimalipuissa.

Mifare DESFire -tunnisteita käytetään paljon myös perinteisiin kulunvalvontasovelluksiin. Esimerkiksi jossakin isossa yrityksessä osalla henkilökunnasta on eri kulkuoikeudet tiettyihin tiloihin. Näiden henkilöiden tunnisteisiin voidaan perusoikeuksien lisäksi asettaa oikeudet päästä näihin tiloihin. Yrityksen henkilöstö pystyy tarpeen vaatiessa muuttamaan tunnisteen oikeuksia. Tämä tietenkin edellyttää sen, että yrityksellä on tunnisteiden kirjoittamiseen vaadittava laite.

Oulun kaupungilla on käytössään OuluCard-niminen monitoimikortti. Kortin palveluihin kuuluvat muun muassa liikuntaliput sekä kaupunkiliikenteen bussiliput. Kortin toiminta perustuu Mifare DESFire -tekniikkaan. Vuoden 2011 alussa kaupunkikorttijärjestelmää päivitettiin ja sen myötä palveluvalikoima laajenee lähivuosina. (16.)

4 IDESCO OY:N TUOTANTOROBOTTI

Idesco Oy:n tuotannossa on käytössä automaatiorobotti, jolla voidaan automatisoidusti ohjelmoida RFID-tunnisteita (kuva 10). Robotin tärkeimpiin komponentteihin kuuluvat ohjelmoitava logiikkaohjain, RFID-lukija sekä Linx-leimasin. Lukija ja leimasin on kytketty sarjaportin kautta PC-tietokoneeseen, jossa sijaitsee tunnisteiden ohjelmointia kontrolloiva käyttöliittymä.



KUVA 10. Idescon tuotannossa oleva automaatiorobotti

Tunnisteiden ohjelmointiprosessi lähtee liikkeelle siitä, että avataan PC:ltä ohjelmoitaville tunnisteille tarkoitettu käyttöliittymäsovellus, alustetaan robotin parametrit, kytketään robotille oikea lukijalaite, laitetaan Linx-leimasin valmiuteen ja asetetaan haluttu määrä tunnisteita syöttömakasiiniin. Kun tarvittavat asetukset on tehty, käynnistetään robotin liukuhihna ja laitetaan käyttöliittymä odottamaan robotilta saapuvaa signaalia.

Tunniste siirtyy liukuhihnaa pitkin ja pysäyttimen avulla se jää lukijan alle ohjelmoitavaksi. Liukuhihnan päälle asetettu anturi havaitsee lukijalle menevän tunnisteen ja lähettää käyttöliittymäsovellukselle käskyn siitä, että tunniste voidaan ohjelmoida. Käyttöliittymä lähettää ohjelmointikomennon lukijalle ja lukija ohjelmoi tunnisteen. Kun ohjelmointi on suoritettu, käyttöliittymä komentaa

lukijan lukemaan tunnisteesta juuri kirjoitetetun tiedon, jotta voidaan varmistua, että ohjelmointi onnistui. Mikäli ohjelmointiprosessi onnistui, käyttöliittymä lähettää robotille signaalin, jolloin liukuhihnalla sijaitseva pysäytin nousee ja tunniste jatkaa matkaansa. Jos ohjelmointi ei onnistunut, odotetaan robotin logiikkaan asetettu aika, jonka jälkeen pysäytin nousee ja tunniste ohjataan hylkykoriin.

Seuraavaksi tunniste kulkee leimasimen alta, jossa tunnisteen pintaan leimataan jokin ennalta määritetty koodi. Nyt tunniste on valmis ja se voidaan siirtää vastaanottomakasiiniin.

4.1 Robotin ohjelmoitava logiikkaohjain

Ohjelmoitava logiikkaohjain eli PLC (Programmable Logic Controller) on pieni tietokone, jota käytetään esimerkiksi tehtaan kokoamislinjan ohjaamisessa. Yhdellä PLC:llä pystytään korvaamaan satoja tai jopa tuhansia aiemmin käytettyjä releitä tai ajastimia. (17.)

Ohjelmoitava logiikka on mikroprosessoripohjainen laite, joka sisältää tulo- ja lähtöportteja. Portteihin voidaan kytkeä esimerkiksi erilaisia antureita, moottorin käynnistimiä, solenoideja, merkkivaloja ja venttiilejä. Logiikka ohjaa portteja PLC:n muistiin sijoitetun ohjelman ja sensoreiden antamien tietojen mukaan. Tulot ja lähdöt voivat olla joko digitaalisia tai analogisia. Digitaalisten tulojen ja lähtöjen signaalit ilmaisevat ainoastaan, onko signaali 1 (tosi) vai 0 (epätosi). Analogiset signaalit käyttäytyvät kuten äänenvoimakkuuden säätimet. Ne välittävät kaikki arvot nille määritetyltä toiminta-alueelta. Tyypillisiä analogisten signaalien avulla lähetettyjä mittaustietoja ovat esimerkiksi paine-, virtaus- ja lämpötilatieto. (17.)

Ohjelmoitavan logiikan ohjelma voidaan tehdä monella eri tavalla. Ohjelmointikielenä voidaan käyttää tikapuulogiikalla toimivia kieliä (esimerkiksi Ladder Diagram) tai perinteisiä ohjelmointikieliä, kuten esimerkiksi C-kieli. Ohjelmoitava logiikka suorittaa ohjelmassa määritetyt tehtävät. Ohjelmaan on voitu asettaa joitakin muutettavia parametreja, joiden hallintaan tarvitaan jonkinlainen käyttöliittymä. (17.) Kuvassa 11 näkyy Idescon robotin käyttöliittymänä toimiva operointipaneeli.

KUVA 11. Idescon robotin operointipaneeli

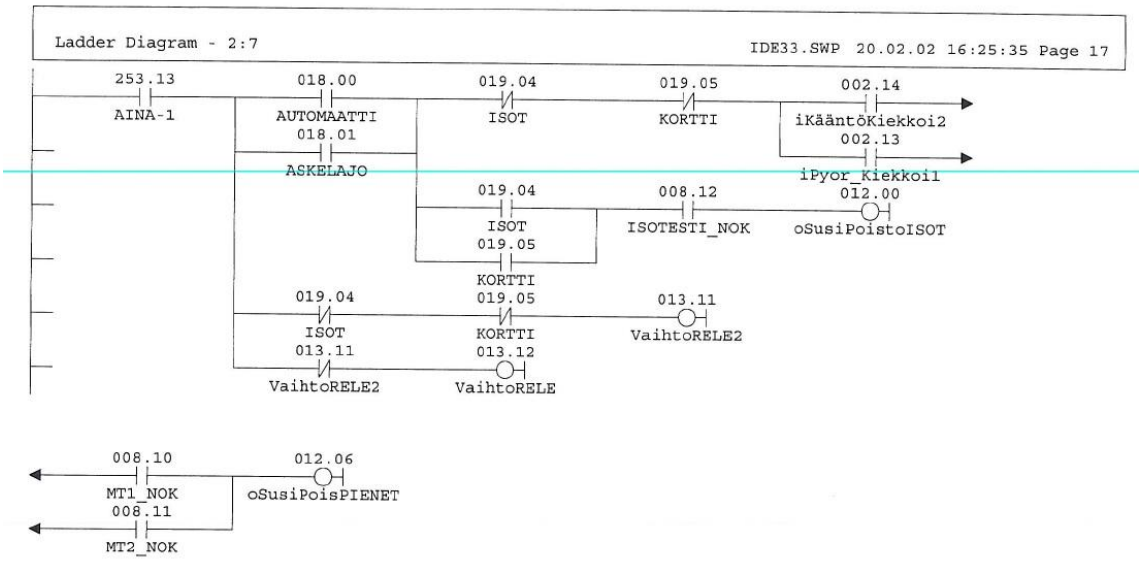
4.2 Ladder Diagram -ohjelmointi

PLC:n ohjelmointiin voidaan käyttää Ladder Diagram -ohjelmointikieltä eli niin sanottua tikapuuohjelmointia. Sitä voidaan myös kutsua relekaavioksi, koska se on alun perin lähtöisin releillä toteutettujen ohjauksien dokumentoinnista. Ohjelmointikielen ideana on se, että ohjelma on kuin tikapuu, jossa jokainen askelma on yhden tai useamman sisääntulon tai ehdon ja yhden ulostulon tai tuloksen välillä. (18, s. 5.) Jokaiselle PLC:n tulolle ja lähdölle pitää antaa oma muistipaikan numero sekä toimintoa kuvaava symboli (taulukko 3).

TAULUKKO 3. PLC:n tulojen ja lähtöjen määritys (18, s. 3)

Muistipaikan numero	Tyyppi	Symboli	Kuvaus	Muodostaja
000.00	Tulo	LS2	hylly kohdalla	rajakytkin
000.01	Tulo	LS1	0-hylly kohdalla	rajakytkin
000.02	Tulo	NEXT_SHELF	seur. hylly halutaan	käyttäjä (nappi)
000.03	Tulo	PREV_SHELF	edell. hylly halutaan	käyttäjä (nappi)
000.05	Tulo	LS3	takaluukku kiinni	turvakytkin
001.00	Lähtö	MC1	moottori taaksepäin	logiikan ohjelma
001.01	Lähtö	MC2	moottori eteenpäin	logiikan ohjelma
004.05	Tulo	MANUAL	käsiohjaus-tila	käyttäjä (kytkin)
002.00	Tulo	ROBOT_TAKE	robotti purkaa	robotin ohjelma
002.01	Tulo	ROBOT_FILL	robotti täyttää	robotin ohjelma
002.02	Tulo	ROBOT_HAND	robotin käsi välissä	robotin ohjelma
003.00	Lähtö	SHELF_OK	haluttu hylly kohdalla	logiikan ohjelma

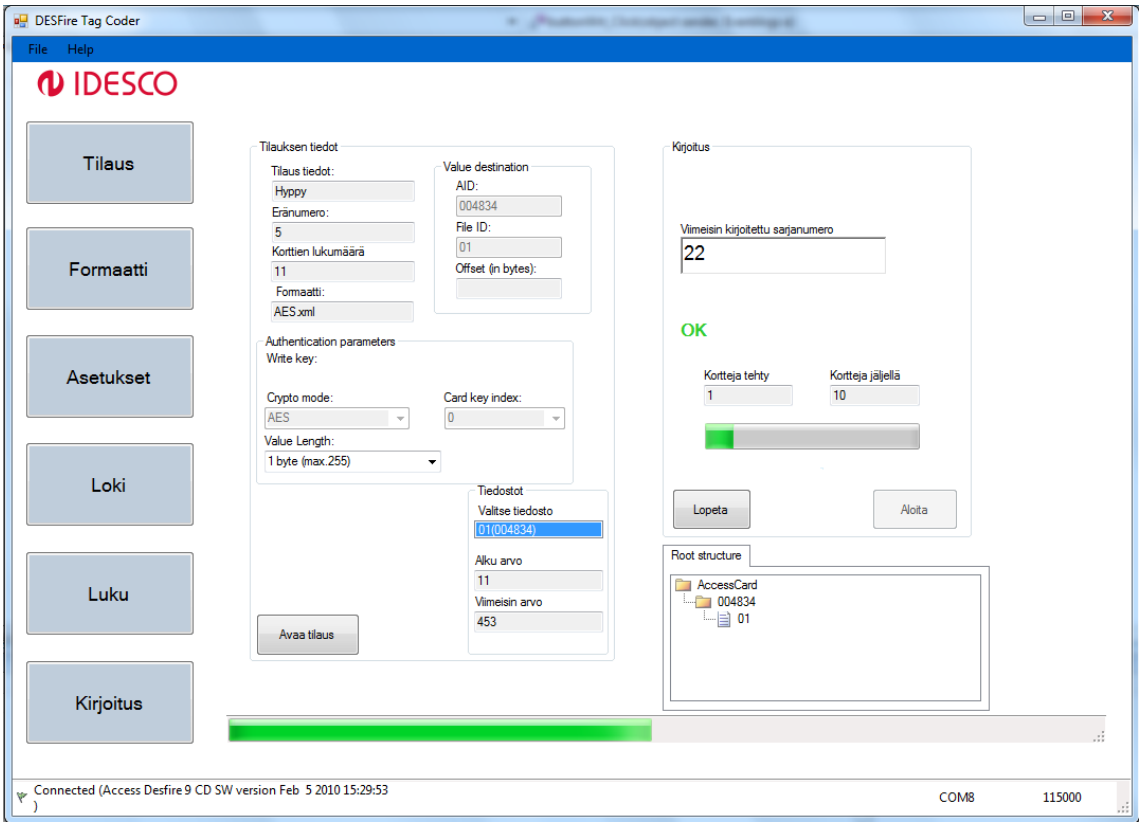
Kuvassa 12 on pieni pala Idescon tuotantorobotin logiikkakaaviota. Siinä kuvataan muun muassa se, millä perusteella vialliset tai ohjelmoinnissa epäonnistuneet tunnisteet laitetaan hylkyyn (oSusiPoistoISOT). OSusiPoistoISOT-ulostuloa ohjataan seuraavilla ehdoilla: AINA-1-kytkin (253.13) on kytketty ja liukuhihna on automaatti- (018.00) tai askelajolla (018.01) ja robotin operointipaneelista on valittu tuotteeksi ISOT (019.04) tai KORTTI (019.05) ja tunnisteen ohjelmointi ei onnistunut (eli ISOTESTI_NOK on aktiivinen). Jos kaikki edellä mainitut ehdot toteutuvat, asetetaan oSusiPoistoISOT-ulostulo aktiiviseksi.



KUVA 12. Idescon tuotantorobotin Ladder Diagram -kaavio

4.3 Käyttöliittymä

Käyttöliittymä on sovellus, jolla tässä tapauksessa hoidetaan tunnisteiden ohjelmointi. Kuvassa 13 näkyy Mifare DESFire -tunnisteiden ohjelmointiin käytetyn käyttöliittymän Kirjoitus-välilehti.



KUVA 13. Käyttöliittymä, jolla voidaan ohjelmoida Mifare DESFire -tunnisteita

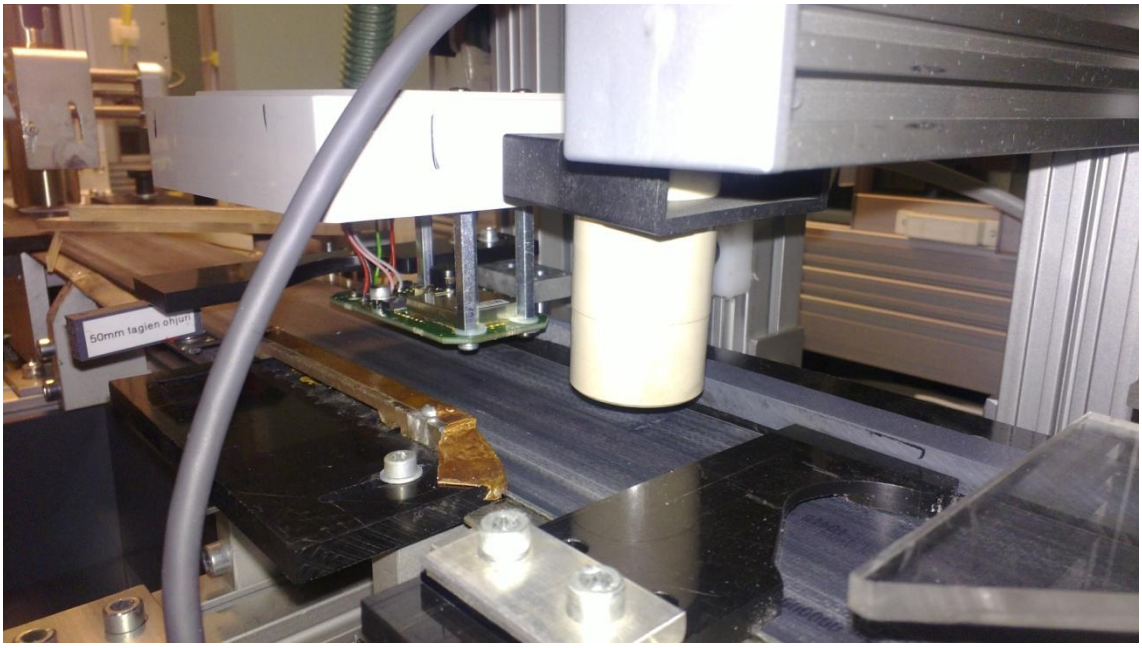
Mifare DESFire -tunnisteiden ohjelmointiin käytetty käyttöliittymä koostuu kahdesta osasta: käyttäjälle näkyvästä osasta eli niin sanotusta GUI:stä (Graphical User Interface) ja tiedonsiirtorajapinnasta, joka toimii toiminnan taustalla. GUI:ssa määritetään muun muassa tunnisteelle tuleva datarakenne eli sovellukset ja tiedostot, tiedostoihin ohjelmoitava data sekä tunnisteen käyttämät avaimet. Käyttöliittymän lähettämistä ja vastaanottamista signaaleista vastaa tiedonsiirtorajapinta. Sen avulla muun muassa alustetaan sarjayhteydet lukijalle ja leimasimelle, välitetään kirjoitus- ja lukukomennot lukijalle, lähetetään tarvittaessa leimasimelle dataa sekä vastaanotetaan ja lähetetään robotin ohjaussignaaleja.

4.4 Lukija

Lukijan tärkein tehtävä tässä sovelluksessa on fyysisesti ohjelmoida tunniste ja lukea tunnisteen sisältöä. Termi lukija voi tässä tapauksessa olla hieman harhaanjohtava, koska tässä sovelluksessa lukijalla ei pelkästään lueta vaan

myös kirjoitetaan dataa. Laitetta kutsutaan kuitenkin lukijaksi, koska yleensä sitä käytetään pelkästään RFID-tunnisteiden lukemiseen.

Tässä sovelluksessa lukijaa käskytetään PC-käyttöliittymän avulla. Käyttöliittymän tiedonsiirtorajapinta lähettää komennon lukijalle sarjaportin kautta. Lukija tulkitsee komennon ja suorittaa sen mukaisen toiminnon. Lukija sisältää Mifare DESFire -piirin, joka keskustelee vastaavan tunnisteella sijaitsevan piirin kanssa. Esimerkiksi kaikki DESFire-teknologiaan liittyvät salausoperaatiot toteutetaan lukijassa. Lukijat alkavatkin olla tänä päivänä toiminnoiltaan melko kehittyneitä. Tuotantorobotin liukuhihnalla käytetään Idesco Oy:n valmistamaa Access 9CD -lukijaa (kuva 14).



KUVA 14. Liukuhihnalla oleva Access 9CD -lukija (vihreä piirilevy)

4.5 Linx-leimasin

Linx-muistesuihkuleimasinta käytetään tunnisteiden leimaamiseen. Tunnisteeseen kirjoitetun leiman tarkoituksena on helpottaa asiakkaan toimintaa lisättäessä uusia tunnisteita RFID-järjestelmään. Jos leimaa ei olisi, pitäisi asiakkaan aina ensin lukea tunnisteen sisältö, jota asiakas voisi käyttää yksilöimään sen, mikä tunniste kuuluu kenellekin henkilölle. Leiman avulla asiakas näkee heti, mikä koodi asetetaan kenellekin. (19.)

Leimasin koostuu leimausyksiköstä, leimauspäästä sekä ohjausanturista. Leimausyksikössä määritetään muun muassa, mihin kohtaan leimattavassa kohteessa leima laitetaan, mitä kohteeseen tullaan leimaamaan sekä minkä kokoisia leimattavat merkit ovat (kuva 15).

KUVA 15. Idescon robottiin liitetty Linx-leimasin

Leimausyksikkö lähettää leimaustiedon leimauspäälle, joka hoitaa fyysisen leimauksen. Jotta leimausyksikkö tietää, milloin kohde on leimattava, tarvitaan ohjausanturia. Tunnisteiden ohjelmointisovelluksessa ohjausanturi ja leimauspää sijaitsevat peräkkäin liukuhihnan päällä ja kun tunniste ohittaa anturin, anturi välittää tiedon leimausyksiköille ja leimausyksikkö käskee leimauspään leimata tunnisteen (kuva 16).

KUVA 16. Robotin liukuhihnalla sijaitsevat ohjausanturi (keltainen pieni kotelo) sekä leimauspää (metallinen lieriö)

Leimasimessa voidaan käyttää kahta eri toimintamenetelmää. Leimasimeen voidaan asettaa jokin kiinteä lähtöarvo, joka tunnisteeseen leimataan. Tämän lisäksi laitteeseen määritetään, kuinka monta leimaa tullaan tekemään (eli kuinka monta tunnistetta ollaan ohjelmoimassa). Aina kun tunniste ohittaa ohjausanturin, leimausyksikköön asetettua arvoa kasvatetaan yhdellä ja seuraavaan tunnisteeseen leimataan siis yhtä isompi arvo.

Toisessa menetelmässä leimausyksikköön ei aseteta mitään alkuarvoa vaan se asetetaan vastaanottamaan dataa sarjaportista. Tässä tapauksessa PC:llä oleva käyttöliittymä lähettää leimasimelle leimattavan datan sarjaportin kautta välittömästi tunnisteen ohjelmoinnin jälkeen. Leimattava tieto voi olla esimerkiksi tunnisteelle kirjoitettu tieto tai osa siitä.

5 TOTEUTUS

5.1 Lähtötilanne

Lähtötilanteessa Idescon tuotannossa olevalla automaatiorobotilla oli mahdollista valmistaa kahdella eri tekniikalla toimivia RFID-tunnisteita, mutta siitä puuttui tuki Mifare DEDFire -tekniikalla toimivilta tunnisteilta. Isojen valmistusmäärien vuoksi oli järkevää, että tunnisteiden valmistamista tehostetaan automatisoidummalla ratkaisulla.

Jotta robotilla pystyttäisiin valmistamaan DESFire-tunnisteita, piti robottiin kytketylle tietokoneelle kehittää käyttöliittymäsovellus. Käyttöliittymän toteutus jaettiin kahteen osaan, joista toinen osa käsitti käyttöliittymän käyttäjälle näkyvän osan toteutuksen ja toinen osa robotin ja käyttöliittymän välisen rajapinnan toteutuksen. Tämän opinnäytetyön tehtävänä oli toteuttaa robotin ja käyttöliittymän välinen tiedonsiirtorajapinta. Käyttöliittymän käyttäjälle näkyvän osan toteutus oli toisen opiskelijan opinnäytetyön aiheena.

5.2 Vaatimusmäärittely

Projektin alussa pidettiin vaatimusmäärittelypalaveri, jossa listattiin käyttöliittymältä vaadittavia ominaisuuksia. Vaatimukset liittyivät suoranaisesti käyttöliittymän käyttäjälle näkyvään osioon eivätkä niinkään tiedonsiirtorajapintaan, mutta toki käyttöliittymän käyttäjälle näkyvän osion vaatimukset sanelivat myös osaltaan sitä, millainen tiedonsiirtorajapinnasta pitäisi tehdä. Mifare DESFire -tunnisteiden joustavan muistirakenteen ansiosta tunnisteen muistiin voidaan kirjoittaa dataa hyvin monella tavalla. Vaatimusmäärittelyssä tehtiin asiakastarpeen ja opinnäytetyöhön käytössä olevan ajan mukaan pientä rajausta siitä, mitä kirjoitustapoja käyttöliittymän pitäisi pystyä tukemaan. Yksi vaatimus oli se, että käyttöliittymän tuli olla käyttäjän kannalta mahdollisimman helppo käyttää. Ehkä tärkein vaatimus oli se, että käyttöliittymän piti varmistua siitä, että tunnisteelle kirjoitettu tieto on onnistunut eikä näin ollen käyttäjän tarvitse manuaalisesti varmistaa kirjoituksen onnistumista. Käyttöliittymän piti pystyä kommunikoimaan robotin, lukijan ja Linxin kanssa. Tiedonsiirtorajapinnan vaatimuksina oli mahdollistaa nämä

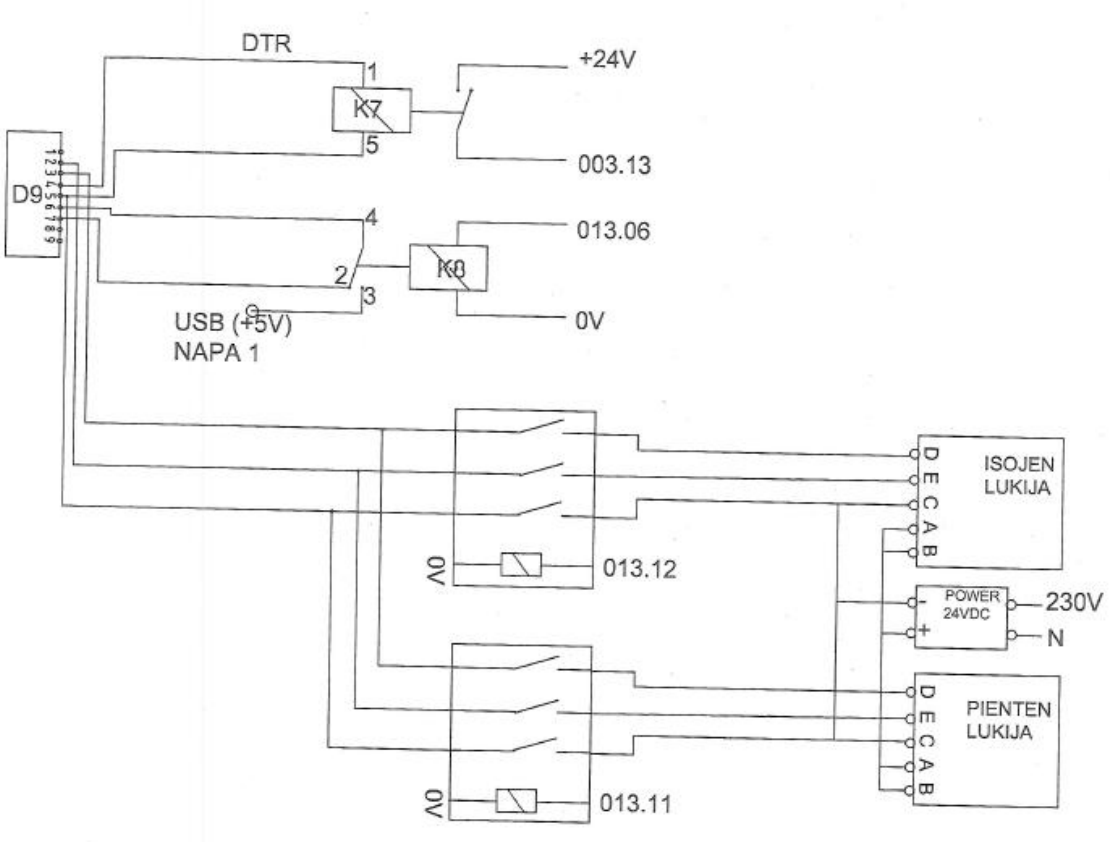
tehtävät. Rajapinta piti toteuttaa siten, että se voidaan liittää osaksi muuta käyttöliittymää.

5.3 Esiselvitys

Kun vaatimukset oli mietitty, ryhdyttiin tekemään esiselvitystä siitä, miten tiedonsiirtorajapinta tulisi toteuttaa. Robottien aikaisemman tuntemuksen puutteen vuoksi työn aloittaminen oli melko haasteellista.

Ensimmäiseksi tutustuttiin robotin toimintaan. Idescon tuotannossa oli mahdollista päästä seuraamaan, miten aiemmat käyttöliittymäsovellukset suorittavat tunnisteiden ohjelmoinnin ja miten robottia käskytetään. Työn selvityksessä liikkeelle lähtöä hidasti se, että aiemmat käyttöliittymäsovellukset toteuttanut henkilö ei enää työskennellyt Idescolla, joten avun saanti robotin logiikkaan liittyen oli vähäistä. Idescolla on robotin valmistajan toimittama pari sataa sivua paksu kansio, jossa on kuvattu robotin toimintalogiikka. Kansion sisällön numerosarjat ja tikapuulogiikat näyttivät aluksi hyvin epämääräisiltä ja vasta robotin fyysisten kytkentöjen selvittämisen jälkeen rupesi robotin toimintalogiikka hahmottumaan. Selvisi, että robotin toimintaa ohjaa ohjelmoitava logiikkaohjain, jonka robotin valmistaja oli ohjelmoinut Ladder Diagram -ohjelmointikielellä.

Tietokoneen sarjaporteista lähtevistä kahdesta kaapelista toinen menee Linx-leimasimelle ja toinen robotin ohjausreleiden kautta lukijalle. Tässä vaiheessa oli selvää, että käyttöliittymän ei tarvinnut ohjata täysin robotin toimintaa, vaan sen piti ainoastaan vastaanottaa tieto siitä, milloin tunnistetta pitää ruveta ohjelmoimaan, sekä lähettää robotille tieto tunnisteen ohjelmoinnin onnistumisesta. Kuvassa 17 näkyy tietokoneen toiseen sarjaporttiin kytketty RS232-liitin (D9) sekä paikka, mihin liittimestä lähtevät johtimet menevät.



KUVA 17. RS232-liittimen kytkentä

Kuvassa näkyviä signaaleja 003.13 ja 013.06 käytetään robotin kanssa kommunikoimiseen ja isojen tai pienten lukija valitaan robotin operointipaneelin kautta.

Kun robotin toimintalogiikka oli selvitetty, ryhdyttiin selvittämään, voidaanko aiemmista käyttöliittymäsovelluksista sekä muista Idescon sovelluksista hyödyntää joitakin osioita. Tämän lisäksi piti miettiä, millä ohjelmointikielellä tiedonsiirtorajapinta tulisi toteuttaa. Idescon tuotevalikoimasta löytyvällä Idesco DESCoder -ohjelmalla voidaan ohjelmoida DESFire-tunnisteita ja ohjelmasta voitiin hyödyntää tunnisteen ohjelmoimiseen vaadittavat funktiot. DESCoder-ohjelmassa tunnisteen ohjelmointifunktioista oli tehty C++-kirjasto. Vaatimusmäärittelypalaverissa oli tullut esille, että käyttöliittymä toteutetaan C#-ohjelmointikielellä, joten DESCoder-ohjelman käyttämä C++-kirjasto oli mahdollista liittää osaksi uutta käyttöliittymää.

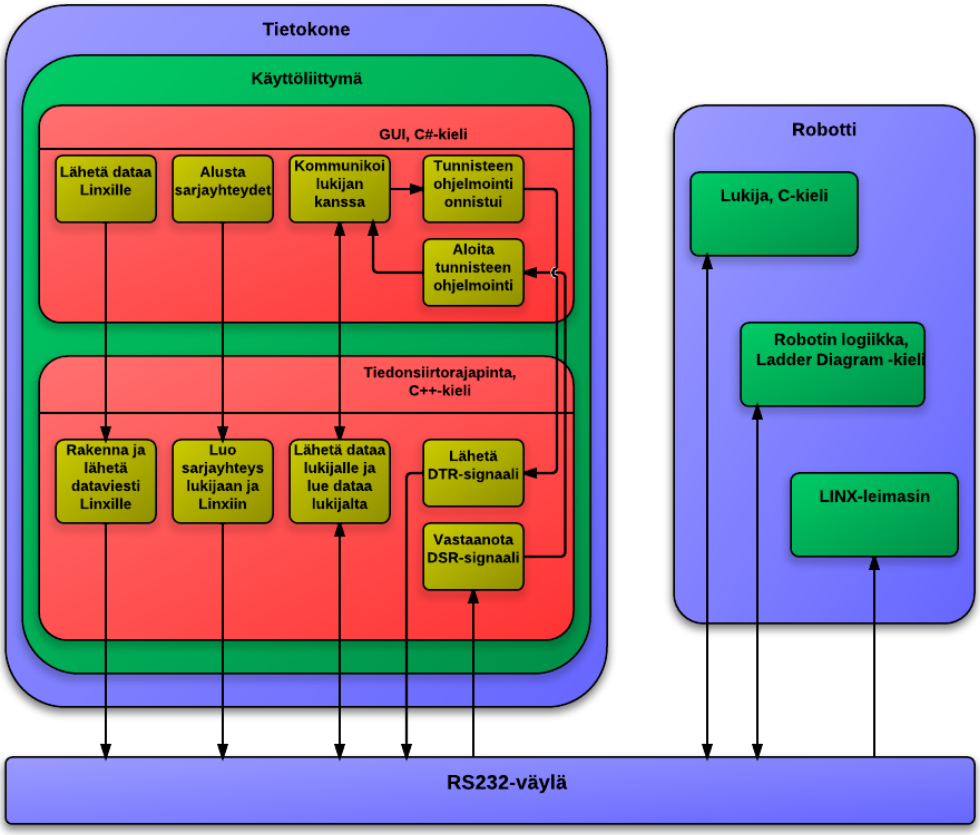
Yksi selvitettävistä asioista oli tutkia Linx-leimasimen kommunikointitapaa. Vanhoista käyttöliittymäsovellusten lähdekoodeista löytyi Linx-leimasimen

kommunikointiin liittyviä funktioita. Funktioiden toiminta näytti hieman epämääräiseltä, joten Linx-leimasimen kanssa kommunikoimisesta sarjaportin välityksellä piti etsiä lisätietoa internetistä. Selvisi, että Linxille lähetettävän viestin tulee olla seuraavanlaisessa muodossa: ESC+STX+komento ID+DATA+ESC+ETX+tarkiste (20, s. 23). ESC (Escape Character), STX (Data start delimiter) ja ETX (Data end delimiter) ovat ASCII-merkistön ohjausmerkkejä. ESC tarkoittaa keskeytysmerkkiä, STX tarkoittaa datan aloituserotinta ja ETX datan lopetuserotinta. (20, s. 29.) ID tarkoittaa lähetettävän viestin tyyppiä. Tässä sovelluksessa ID:nä käytettiin heksa-arvoa 1D, joka tarkoittaa datan latausta Linxiin (20, s. 45). Data-kenttään tulee lähetettävän datan pituus sekä varsinainen data. Viestin loppuun lasketaan tarkiste.

5.4 Kehitys

Tiedonsiirtorajapinnan kehitys tehtiin Microsoft Visual Studio 2008 -kehitysympäristössä. Kehitysympäristössä luotiin uusi DLL (Dynamic Link Library) -projekti, johon ohjelmaa ruvettiin toteuttamaan. Projektiin kopioitiin tunnisteen ohjelmointiin liittyvät funktiot DESCoder-ohjelmasta. DESCoder-ohjelman funktioista löytyi myös lukijan sarjaportin alustukseen liittyvä funktio. Koska käyttöliittymä tarvitsee toimiakseen kahta sarjaporttia, yhden lukijalle ja yhden Linx-leimasimelle, piti kirjastoon toteuttaa toinenkin sarjaportin alustusfunktio.

Tämän jälkeen toteutettiin robotin ohjaukseen liittyviä funktioita. Kuvassa 18 on kuvattu karkeasti se, miten tiedonsiirtorajapinta kommunikoi käyttöliittymän käyttäjälle näkyvän osion ja robotin eri elementtien välillä.



KUVA 18. Tiedonsiirtorajapinnan toiminta tunnisteiden ohjelmointiprosessissa

Robotilta käyttöliittymään päin signaloidaan RS-232-väylän DSR (Data Set Ready) -signaalin avulla. Kun tunniste ohittaa robotin liukuhihnalla olevan anturin, robotti lähettää käyttöliittymälle tiedon RS-232-väylää pitkin. Tiedonsiirtorajapinnan DSR-signaalin vastaanottofunktiossa kuunnellaan, onko DSR-signaali asetettu robotin toimesta aktiiviseksi, ja jos on, ilmoitetaan käyttöliittymälle, että tunnisteen ohjelmointi voidaan aloittaa.

Robotille päin signaloidaan RS232-väylän DTR (Data Terminal Ready) -signaalia käyttämällä. Kun käyttöliittymä on käskenyt lukijan ohjelmoimaan tunnisteen, se välittömästi lukee ohjelmoidun tiedon. Jos luettu tieto vastasi ohjelmoitua tietoa, käyttää tiedonsiirtorajapinnan DTR-funktio DTR-signaalia aktiivisena, jonka jälkeen se palautetaan ei-aktiiviseksi. Tämä on merkkinä robotin logiikalle siitä, että tunniste voidaan päästää jatkamaan matkaa liukuhihnalla.

Datan lähetys Linx-leimasimelle toteutetaan tiedonsiirtorajapinnassa sijaitsevassa funktiossa. Funktiossa rakennetaan käyttöliittymän käskemän datan perusteella Linxille lähetettävä viesti. Viestin rakentamiseen kuuluu funktioon lähetettävän datan järjestäminen, viestin alku- ja loppumerkkien lisääminen sekä viestin sisällöstä laskettavan tarkisteen lisääminen. Kun viesti on rakennettu, se lähetetään sarjaportin kautta Linx-leimasimelle. Käyttöliittymän puolelle toteutettiin C#-kielellä funktio, joka kutsuu C++-kirjastossa sijaitsevaa Linx-datan rakennusfunktiota. Tämä funktio laskee Linxille lähetettävän viestin pituuden ja lähettää datan C++-kirjaston funktiolle. Lisäksi funktiossa käsitellään se, onko leimattava data desimaali- vai heksamuodossa.

Kun tarvittavat funktiot oli saatu toteutettua, ohjeistettiin käyttöliittymää toteuttavaa opiskelijaa siitä, miten C++-kirjasto voidaan integroida osaksi C#-käyttöliittymää sekä miten funktioita voidaan kutsua käyttöliittymästä.

5.5 Testaus

Työn testaus koostui kahdesta osasta: tiedonsiirtorajapinnan testaus testiohjelman avulla sekä varsinaisen käyttöliittymän testaus.

Koska käyttöliittymä ei ollut vielä valmis tiedonsiirtorajapinnan valmistuttua, toteutettiin C#-kielellä yksinkertainen testiohjelma, jolla pystyttiin testaamaan rajapintaa. Tiedonsiirtorajapinta liitettiin testiohjelmaan ja testiohjelma asennettiin robotin tietokoneelle. Testiohjelmalla voitiin testata seuraavat asiat: sarjayhteyden muodostuminen lukijaan ja Linx-leimasimeen, robotin ohjaussignaalien toimivuus sekä datan lähetys Linx-leimasimelle.

Testaus lähti liikkeelle robotin konfiguroimisesta. Kun robotti oli konfiguroitu, käynnistettiin tietokoneelta testiohjelma. Testiohjelma otti automaattisesti yhteyttä lukijaan ja leimasimeen käynnistymisen yhteydessä. Kun robotin liukuhihnan päällä olevan anturin alta meni tunniste, piti testiohjelman pystyä tulkitsemaan se. Kun tunniste ohitti anturin, laskeutui liukuhihnalle pysäytin, jolla tunniste saatiin pysähtymään lukijan alle. Testiohjelmaan toteutettiin näppäin, jota painettaessa robotille lähetettiin signaali, jolloin pysäyttimen piti nousta. Mikäli näppäintä ei painettu ja robotin operointipaneeliin määritetty odotusaika

umpeutui, tunniste ohjattiin hylkyyn. Linxille lähetetyn datan perille meno varmistettiin siten, että Linx asetettiin datan vastaanottotilaan ja testiohjelmalla lähetettiin Linxille data, jonka jälkeen Linxin näytölle piti ilmestyä lähetetty data. Muutaman testikierroksen ja testiohjelman korjauksen jälkeen kaikki nämä toiminnot saatiin toimimaan.

Kun varsinainen käyttöliittymä oli valmis ensimmäisiin testeihin, se asennettiin robotin tietokoneelle. Käyttöliittymän testaus toteutettiin yhdessä käyttöliittymän valmistaneen opiskelijan kanssa. Aluksi ongelmia tuli paljon, mutta korjaavien toimenpiteiden jälkeen niistä selvittiin. Idescon tuotannosta saatiin käyttöön tyhjiä DESFire-tunnisteita, joihin yritettiin ohjelmoida tietoa käyttöliittymän avulla. Robotilla ajettiin useita testejä, joissa tunnisteiden muistiin kirjoitettiin eri määrä sovelluksia ja tiedostoja sekä tiedostoihin kirjoitettua dataa vaihdeltiin. Käyttöliittymästä testattiin myös se, että se pystyy kirjoittamaan tunnisteelle sekä AES- että 3DES-salattua dataa.

Jos asiakkaalla on rikkoutunut tai hävinnyt tunnisteita ja asiakas haluaa tilata uudet tunnisteet, joissa on sama data kuin aiemmin, tunnisteiden datat voidaan kirjoittaa alekkain Excel-tiedostoon, josta käyttöliittymä osaa poimia tunnisteeseen kirjoitettavan datan. Tässä tapauksessa tunnisteelle kirjoitettu data yleensä lähetetään käyttöliittymältä Linx-leimasimelle. Tilanne testattiin käyttöliittymällä ja tulos oli positiivinen. Tiedonsiirtorajapinnan muutkin ominaisuudet toimivat käyttöliittymän kanssa erinomaisesti.

Käyttöliittymän toiminta testatiin lopuksi yhdessä käyttöliittymää jatkossa käyttävän tuotannon henkilön kanssa. Testeissä käytiin käyttöliittymän toiminta vielä pääpiirteittäin läpi, jonka jälkeen tuotannon henkilö antoi hyväksynnän käyttöliittymälle.

6 YHTEENVETO

Työn tavoitteena oli toteuttaa tiedonsiirtorajapinta, joka yhdessä käyttöliittymäsovelluksen kanssa mahdollistaa Idesco Oy:n tuotannossa olevalla automaatiorobotilla Mifare DESFire -tunnisteiden valmistamisen. Lopputuloksena tämä tavoite toteutui ja saimme yhdessä Ville Koivumäen kanssa toteutettua toimivan käyttöliittymäsovelluksen. Käyttöliittymää tullaan jatkossa käyttämään Idescon tuotannossa.

Automatisoidulla RFID-tunnisteiden ohjelmoinnilla saavutetaan merkittävä etu ensinnäkin nopeudessa ja sitä kautta myös kustannuksissa verrattuna siihen, että jokainen tunniste ohjelmoitaisiin käsin. Mifare DESFire -tekniikan yleistyessä ja sitä kautta tunnisteiden tilausmäärien noustessa on automatisoidusta ratkaisusta varmasti merkittävä hyöty.

Opinnäytetyössä valmistettua tiedonsiirtorajapintaa voidaan tulevaisuudessa hyödyntää uusien RFID-tekniikoiden tuen liittämisessä automaatiorobottiin. Lisäksi projektissa valmistuneesta käyttöliittymästä saadaan varmasti hyvä pohja uusien käyttöliittymien suunnitteluun.

7 POHDINTA

Työn alussa liikkeelle lähtö oli hidasta, koska aihe oli hyvin tuntematon ja minulla ei ollut aiempaa kokemusta automaatioroboteista. Sinnikkään tutkiskelun jälkeen pääsin työssäni eteenpäin ja tuloksia rupesi syntymään. Liikkeelle lähtöä edisti huimasti myös se, että Idescon työskentelytavat olivat tulleet jo tutuksi edelliseltä kesältä eikä niiden opetteluun tarvinnut käyttää aikaa. Kun robotin toimintalogiikka ja robotin ja käyttöliittymän välinen yhteys oli saatu selvitettyä, oli itse ohjelmointi melko suoraviivaista toteuttaa. Mifare DESFire -tekniikalla on mahdollista toteuttaa hyvin paljon erilaisia sovelluksia ja sen takia käyttöliittymän testauksessa piti testata todella paljon eri tilanteita. Se vei aikaa ja käyttöliittymän virheitä jouduttiin korjaamaan useasti. Kaikesta huolimatta ongelmista selvittiin ja tavoitteeseen päästiin.

Näin jälkeenpäin ajateltuna työn vaatimusten määrittämiseen olisi pitänyt käyttää enemmän aikaa ja se olisi pitänyt tehdä tarkemmin, jolloin työn suoritusvaiheessa ei olisi enää tarvinnut miettiä niitä ja työ olisi edennyt suoraviivaisemmin. Työn aihe oli kaiken kaikkiaan hyvin mielenkiintoinen ja opettavainen.

Työssä pääsi tutustumaan automaatiorobotin toimintaan ja sen kytkentäkaavioihin, Ladder Diagram -ohjelmointiin sekä moneen muuhun asiaan, jotka eivät olleet minulle entuudestaan tuttuja. Lisäksi ohjelmointitaitoni olivat ennen tätä projektia keskittyneet pelkästään C-kielen osaamiseen, mutta nyt minulla on perusteet myös C++- ja C#-ohjelmointikielistä. Opinnäytetyön tekeminen kuvasi myös hyvin, mitä eri vaiheita työelämän projektehin kuuluu ja mitä missäkin vaiheessa tehdään.

LÄHTEET

1. Mitä on RFID?. Saatavissa: http://www.rfidlab.fi/rfid-tietoutta. Hakupäivä 6.1.2012.

2. Rinta-Runsala, Esa – Tallgren, Markus 2004. RFID-tekniikan hyödyntäminen asiakkuudenhallinnassa. Saatavissa: http://www.vtt.fi/inf/julkaisut/muut/2004/rfid-raportti.pdf. Hakupäivä 6.1.2012.

3. RFID-tekniikan perusteet. Saatavissa: http://www.rfidlab.fi/rfid-tekniikan-perusteet. Hakupäivä 6.1.2012.

4. RFID-tekniikan käyttämät taajuusalueet. Saatavissa: http://www.rfidlab.fi/rfid-tekniikan-käyttämät-taajuusalueet. Hakupäivä 6.1.2012.

5. Mifare. 2011. Saatavissa: http://en.wikipedia.org/wiki/MIFARE. Hakupäivä 8.1.2012.

6. AN10833 MIFARE Type Identification Procedure. 2011. Saatavissa: http://www.nxp.com/documents/application_note/AN10833.pdf. Hakupäivä 8.1.2012.

7. Finkenzeller, Klaus 2003. RFID Handbook: fundamentals and applications in contactless smart card and identification. Second edition. Munchen: Wiley.

8. MIFARE DESFire EV1. 2010. Saatavissa: http://www.nxp.com/documents/leaflet/75015782.pdf. Hakupäivä 8.1.2012.

9. MF3ICDx21_41_8, MIFARE DESFire EV1 contactless multi-application IC. Saatavissa: http://www.nxp.com/documents/short_data_sheet/MF3ICDX21_41_81_SDS.pdf. Hakupäivä 8.1.2012.

10. PHILIPS, mifare® DESFire Functionality. 2006. Saatavissa: http://read.pudn.com/downloads134/ebook/572228/M306_Mifare_DESFire_Func_V1.pdf. Hakupäivä 8.1.2012.

11. AN10969, System level security measures for MIFARE installations. 2010. Saatavissa: http://www.nxp.com/documents/application_note/AN10969.pdf. Hakupäivä 8.1.2012.

12. Message Authentication Code. 2011. Saatavissa: http://en.wikipedia.org/wiki/Message_authentication_code. Hakupäivä 8.1.2012.

13. Conrad, Eric. Data Encryption Standard. Saatavissa: http://www.giac.org/cissp-papers/62.pdf. Hakupäivä 8.1.2012.

14. Havukainen, Jani – Kansanen, Maiju. 2004. AES, Advanced Encryption Standard. Saatavissa: http://www2.it.lut.fi/kurssit/03-04/010628000/Seminars/AES.pdf. Hakupäivä 8.1.2012.

15. Electronic ticketing. Saatavissa: http://www.nxp.com/applications/automatic-fare-collection/electronic-ticketing.html#design-considerations. Hakupäivä 12.1.2012.

16. OuluCard. Saatavissa: http://www.ouka.fi/kaupunkikortti. Hakupäivä 12.1.2012.

17. Ohjelmoitava logiikka. 2011. Saatavissa: http://fi.wikipedia.org/wiki/Ohjelmoitava_logiikka. Hakupäivä 14.1.2012.

18. AS-0.2230 Automaatio- ja systeemitekniikan laboratoriotyöt, Työ 1: Logiikka ja robotti. Teknillinen korkeakoulu, Automaatiotekniikan laboratorio. Saatavissa: http://automation.tkk.fi/attach/AS-0-2230/Labratyo1_2006.pdf . Hakupäivä 14.1.2012.

19. Saukko, Jaana 2012. Tuotantopäällikkö, Idesco Oy. Haastattelu 2.2.2012.

20. Linx Remote Communications Interface. 2007. Saatavissa: http://www.ptasia.biz/download/ptasia_biz/download%20center/RCI_RefMan_EN_Iss_8.pdf. Hakupäivä 15.1.2012.